20 year

Yahiya hammouri

مدرسة المادة منار نصيرات

[ASP.NET]

نبذة عن المادة:

هي أساسيات تصميم الصفحات والمواقع الديناميكية المطلوبة وسوق العمل ، ويتعرف من خلالها الطالب على كيفية تصميم صفحات تفاعلية، يتم من خلالها ربط الموقع بملفات وقواعد بيانات موجودة على الجهازالمزود.

بسم الله الرحمن الرحيم

مادة : ASP.NET

اعداد : يحيي الحموري

جامعة البلقاء التطبيقية–كلية الحصن الجامعية

الاردن – اربد – بيت راس

قسم تكنولوجيا المعلومات

اهدي هذا الكتاب الى امي وابي الحبيبين واخوتي واقاربي واخواني المسلمين، وارجو ان يعود هذا الكتاب عليكم بالفائدة

Yahya_line@yahoo.com

لغة الدوت نت 2008

اولا :

.Net-

تعني بالانجليزية (Frame Work) أي (اطار العمل)

- وهي اطار عمل لجميع اللغات.

- ظهور الانترنت ادى الى :

أ – تبادل البيانات

ب – البحث عن المعلومات

ت – الاشتراك في المعلومات

- نتيجة لذلك لجأ المبرمجون الى استخدام تطبيقات للمعالجة ، هذه التطبيقات هي عبارة عن تقنيات معالجة لصفحات الويب(Web) على اجهزة الخوادم.

مثال على ذلك . لغة(ASP)

- ASP: تعني: Action Server Page

☒ سلبيات ASP:

أ – تطوير النظم باستخدامASP امر معقد

ب – ليست برمجية بحد ذاتها (استخدم المبرمجون vbScribt)

ت – المشكلة واجهت مطوري نظم الويب في تبادل وتشارك البيانات.

:HTML

- كانت تقليد ، وتهتم بتنسيق صفحات الويب

- استخدمت فيها بروتوكولات TCP/IP.

☒ ايجابيات لغة ASP.Net

أ – يكتب البرنامج مرة واحدة،ويتم تنفيذه على كل منصات العمل.

- مثال : Notebook -Mobile

ب -يمكن ان تعمل على كل مختلف اصدارات نظم التشغيل.

- مثال : Windows = Unex = Lonex

☒ بيئة التطوير المستخدمة في ASP.Net
Web Developer 2008

*** لغات التطوير المستخدمة ***

1. فيجوال ستديو (C#)
2. ويب ديفيلوبير (Web Developer ;2008)

■ بعض المعلومات الهامة:

تحفظ ال web page بامتداد Asp

- عند تنفيذ البرنامج ، تظهر صفحة ويب (صفحة انترنت).
- مثال : النترنت اكسبلورر(Internet Explorer).

= الهدف من هذه المادة ،انشاء صفحات ويب وربطها مع بعضها ،من اجل انشاء Website

- لتشغيل البرنامج :

–اضغط زر البدء start

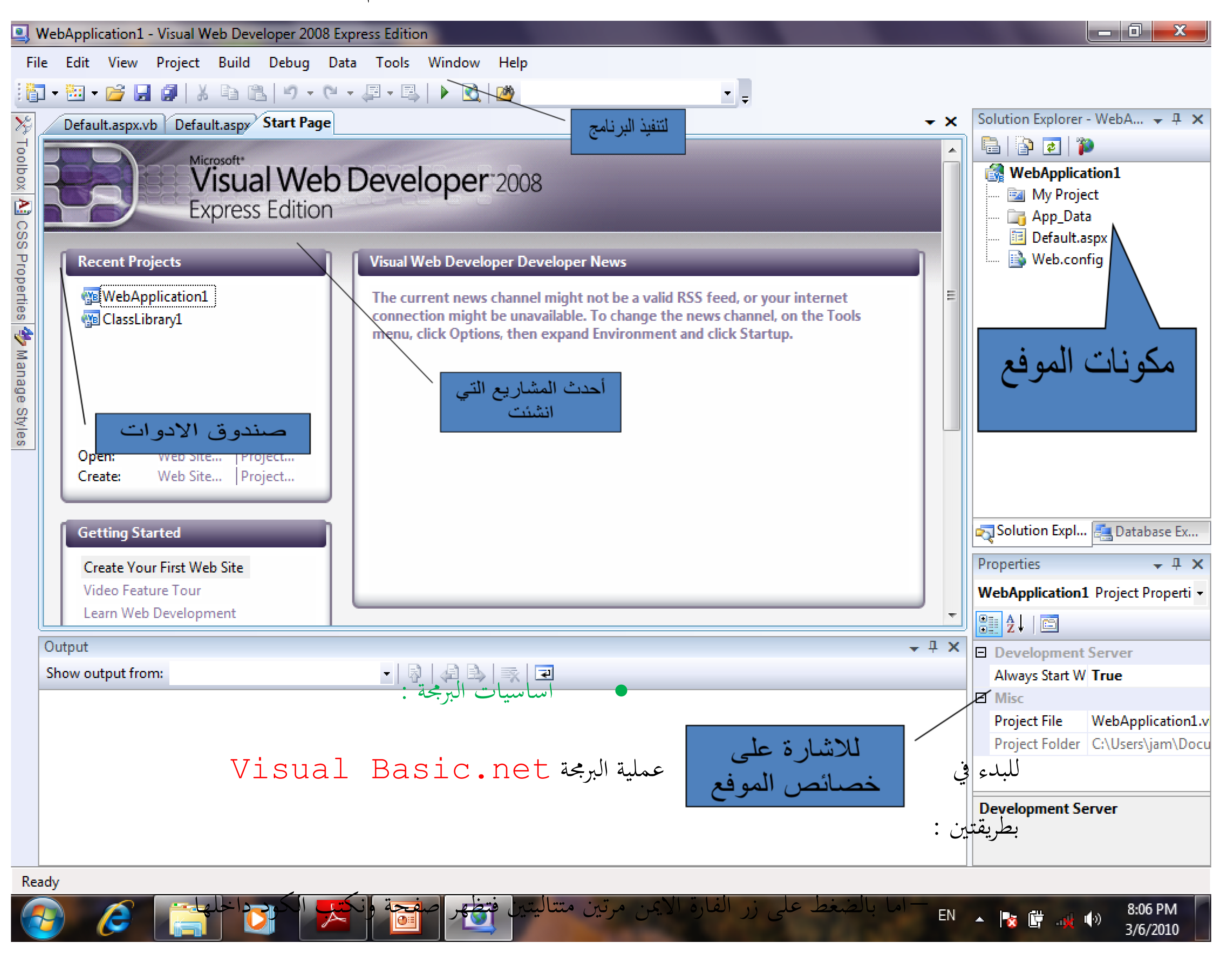
– نختار من البرامج Microsoft Visual web developer2008

- لانشاء صفحة ويب :

– من شريط القوائم Menu bar نختار ملف file ومن ثم موقع جديد new website

– يظهر الصندوق new website ونختار وحدة خدمة الويب الخاصة بلغة ASP.Net (ASP.Net website)

– ونختار اللغة Visual Basic ومكان التخزين ومن ثم OK



- أساسيات البرمجة :

للبدء في عملية البرمجة Visual Basic.net بطريقتين :

– اما بالضغط على زر الفارة الايمن مرتين متتاليتين فتظهر صفحة ونكتب الكود داخلها

–او باستخدام script في ال source داخل صفحة ال HTML

- طرق تعريف المتغيرات :

```
Partial Class _Default
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)

    End Sub
End Class
```

لانشاء صفحات متداخلة :

Website → Add New Item
Web form

لكتابة الكود :

Design → Double Click

تعريف المتغير :

- طرق تعريف المتغيرات :–

أ -نوع المتغير Dim x AS

ب - Dim x As String = NewString

Dim x As String = "HAMMOURI"

Dim x

X = 10

- لتعريف اكثر من متغير :

- Dim x As Long, I , j, k As Integer, s As String

مثال :

```
Dim X As String ="HELLO"
MsgBox (x)
```

- استخدام عبارة Explicit

- Option Explicit Off :- عبارة مسطورة في اعلى الملف ، فيمكنك استخدام المتغيرات والتعامل معها مباشرة دون الحاجة الى التصريح عنها بـ (Dim)

```
Option Explicit Off

Partial Class _Default
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object,
        x = "hjhkjh"
    End Sub
End Class
```

قابلية الرؤية (Visibility وعمر المتغيرات :

- المثال التالي يبين أن المتغير X ، يمكنك الوصول إليه في الإجراء MySub1 () واستخدامه، ولا يمكنك استخدامه من خارج الإجراء:

```
Sub MySub1 ()
 Dim X As Integer
X = 20
End Sub
 Sub MySub2 ()
```

المثال السابق ' X لا يمثل المتغير

```
MsgBox (X)
End Sub
```

- عمر المتغير Lifetime، فتمثل الفترة التي يظل فيها المتغير محتفظا بقيمته، فالمتغير الموجود في الشيفرة السابقة، سينتهي ويفقد القيمة 20 التي كان محتفظا بها بمجرد الانتهاء من تنفيذ MySub1 ()

- أنواع المتغيرات من منظور الرؤية والعمر

1. المتغيرات المحلية الديناميكية (*Dynamic Local Variables*) :

متغيرات يتم الإعلان عنها داخل الإجراءات، وعمر المتغير يبدأ من السطر الذي تصرح فيه عن المتغير وينتهي بعد الانتهاء من تنفيذ الإجراء. أما بالنسبة لقابلية الرؤية فهي محصورة داخل الإجراء الذي صرحت فيه فقط.

- أيضا للتصريح عن متغير محلي ديناميكي. Dim تستخدم الكلمة المحجوزة

```
For counter = 1 To 10

Dim y as integer

...

...
```

متغير حقيقي ديناميكي

```
Next
```

رسالة خطأ

```
x = y '
```

2. المتغيرات المحلية الستاتيكية *(Static Local Variables)* :

هي نفس المتغيرات المحلية الديناميكية، المتغيرات المحلية الستاتيكية لذلك كل ما في النوع السابقة ينطبق هنا دون أي اختلاف، باستثناء أن عمرها الافتراضي ابدي (أي يستمر المتغير الاستكاتيكي محتفظاً بقيمته حتى نهاية البرنامج أو موت الكائن التابع له)

- للتصريح عن متغير ستاتيكي: تستخدم الكلمة المحجوزة Static عوضا عن Dim

```
Sub Counter ()
    Static counter As Integer
    Counter = counter + 1
…
…
End Sub
```

3. المتغيرات على مستوى الوحدة البرمجية والمتغيرات العامة:

- تعرف المتغيرات على مستوى الوحدة باستخدم الكلمة المحجوزة Dim أو Private

```
' متغيرات على مستوى الوحدة
Dim x As Integer

Private y As Integer

Sub Main ()

x = 50
```

...

```
End Sub

Sub Test ()

y = 10

End Sub
```

- أما المتغيرات العامة استخدم الكلمة المحجوزة Public

```
Public x As Integer
```

" **متغير عام**"

```
Sub Main ()

x = 5

End Sub
```

' وحدة برمجية اخرى

```
Sub Test ()
```

x **يمكن الوصول إلى المتغير** ' **1**=

```
End Sub
```

- انواع المتغيرات :

النوع	الحجم	مجال القيمة
Boolean	2 بايت	True (صح) أو False (خطأ).
Byte	1 بايت	عدد صحيح من 0 إلى 255.
Char	2 بايت	حرف واحد من نوع UNICODE.
Date	8 بايت	وقت من الساعة 0:00:00 إلى الساعة 11:59:59، كما يشمل تاريخ من يوم 1 يناير لعام 0001 إلى 31 ديسمبر لعام 9999.
Decimal	16 بايت	عدد صحيح من 0 إلى ±79,228,162,514,264,337,593,543,950,335 أو عشري من 0 إلى ±7.9228162514264337593543950335
Double	8 بايت	عدد عشري من 1.79769313486231570E+308 إلى 4.94065645841246544E-324 بالنسبة للأعداد السالبة. ومن 4.94065645841246544E-324 إلى 1.79769313486231570E+308 بالنسبة للأعداد
النوع	الحجم	مجال القيمة
	4 بايت	
Long	8 بايت	عدد صحيح من – 9,223,372,036,854,775,808 إلى 9,223,372,036,854,775,807.
Object	4 بايت	جميع القيم والأنواع يمكن حفظها هنا.
Short	2 بايت	عدد صحيح من – 32,768 إلى 32,767.
Single	4 بايت	عدد عشري من 3.4028235E+38 إلى -E 1.40129845 بالنسبة للأعداد السالبة. ومن -E 1.40129845 إلى 3.4028235E+38 بالنسبة للأعداد الموجبة.
String	10 + (2 * عدد الحروف) بايت	من 0 إلى 2 مليار حرف من نوع UNICODE.

استخدام التركيبات (Structure) :

- يعرف هذا النوع من التركيبات بالأنواع **المعرفة من قبل المستخدم** ***User Defined Types***
- بحيث تمكنك من دمج أنواع مختلفة من المتغيرات وضمها في تركيب أو كتلة واحدة.
- لتعريف تركيب جديد استخدم الكلمة المحجوزة *Structure*
- مثال:

```
Structure Person
Dim Name As String
Dim Age As Integer
End Structure
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Dim ali As Person
ali.Name = " ahmad "
ali.Age = 69
Response.Write(ali.Name & "          "& ali.Age )
End Sub
```

تمرين : رتب مصفوفة بشكل تصاعدي؟

استخدام التركيبات (Structure):

- يعرف هذا النوع من التركيبات بالأنواع **المعرفة من قبل المستخدم** ***User Defined Types***
- بحيث تمكنك من دمج أنواع مختلفة من المتغيرات وضمها في تركيب أو كتلة واحدة.
- لتعريف تركيب جديد استخدم الكلمة المحجوزة *Structure*
- مثال:

لتعريف التركيب

```
Structure Person

   Dim Name As String

   Dim Age As Integer

End Structure

Protected Sub Page_Load(ByVal sender As
Object, ByVal e As System.EventArgs)

Dim ali As Person

ali.Name = " ahmad "

ali.Age = 69

Response.Write(ali.Name & "        "&
ali.Age )

 End Sub
```

التركيبات المتداخلة (Nested):

```vb
Partial Class _Default
    Inherits System.Web.UI.Page
    Structure Person
        Structure AddressStruct
            Dim City As String
            Dim Countrey As String
        End Structure
        Dim Name As String
        Dim Age As Integer
        Dim Address As AddressStruct
    End Structure
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As Syste
        Dim ali As Person
        ali.Name = " ahmad "
        ali.Age = 69
        ali.Address.City = " Irbid"
        ali.Address.Countrey = " jordan "
        Response.Write(ali.Name & "<br>" & ali.Address.City)
    End Sub
End Class
```

انشاء اقتران واستدعائه داخل تركيب :

```vb
Partial Class _Default
    Inherits System.Web.UI.Page

    Structure Customer
        Dim Name As String
        Dim Income As Decimal
        Dim DeliveryZone As Integer
        Function DeliveryCost() As Decimal
            If DeliveryZone > 3 Then
                Return 25
            Else
                Return CDec(12.5)
            End If
        End Function
    End Structure
    Protected Sub Page_Load(ByVal sender As Object, ByVal e
        Dim buyer As Customer
        buyer.Name = "Martha"
        buyer.Income = 20000
        buyer.DeliveryZone = 1
        MsgBox(buyer.DeliveryCost)
    End Sub
End Class
```

اساسيات فيجوال بيسك دوت نت(VB.Net):

–للبدء في عملية البرمجة Visual Basic.net بطريقتين :

- اما بالضغط على زر الفارة الايمن مرتين متتاليتين فتظهر صفحة ونكتب الكود داخلها
- او باستخدام script في ال source داخل صفحة ال HTML

التعرف على علامة الترميز (Script):

```
<%@ Page Language="VB" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transiti

<script runat="server">
    Sub Page_Load()
        Dim s As String = "welcome"
        Dim x As String = InputBox("enter your name")
        Response.Write(s & "<br>" & x)
    End Sub

</script>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
    <div>
    </div>
    </form>
</body>
</html>
```

الوحدة السادسة :

عمليات برمجة (Object Oriented):

- مثال من الواقع على الكائنات Object
- OOP التعرف على بعض المفاهيم الخاصة بعملية برمجة الكائنات
- انشاء الفئات
- استخدام الفئة في انشاء الكائن
- استخدام الأساليب METHOD/ FUNCTIONS والخصائص PROPERITES المدرجة في الكائنات
- انشاء نسخ متعددة من احدى الفئات
- المتغيرات الخاصة وامكانية الوصول اليها
- الكائنات المدرجة داخل بعضها البعض
- يمكن تحويل **Structure** إلى فئة باستخدام الكلمة المحجوزة **Class**:

```
Structure Person
Dim name as string
Dim Age AS Integer
End Structure
```

```
Class Person
Dim name AS String
Dim Age AS Integer
End Class
```

- نقطة الاختلاف الرئيسية بين الفئات والتركيبات هي أن الفئات من النوع **Reference Type** بينما التركيبات من النوع **Value Type**
- قابلية الوصول لمحتويات الفئة تختلف عن قابلية الوصول لمحتويات التركيبات
- قابلية الوصول الافتراضية في الفئات هي **Private**
- أما مع التركيبات فهي **Public**

مثال :-

انشاء الفئات

- لتعريف الفئة:

```
Class Person
Public Name As String
Public Age As Integer
End Class
```

تعريف متغير يسمى كائن واعطاء القيم للمتغيرات

```
Dim obj As New Person()
 mahmoud " obj.Name = " ali
obj.Age = 99
Response.Write(obj.Name
```

```vb
Partial Class _Default
    Inherits System.Web.UI.Page

    Class Person
        Public Name As String
        Public Age As Integer
    End Class

    Protected Sub Page_Load(ByVal sender As Object,
        Dim ali As New Person()
        ali.Name = " ali  mahmoud "
        ali.Age = 99
        Response.Write(ali.Name)

    End Sub
End Class
```

أما إن أنشأت إجراء(سواء **Function** أو **Sub**) فمحدد الوصول الافتراضي **Public**

```vb
Class Person
Sub PublicMethod () ' Public
End Sub
Public Sub PublicMethod2 () ' Public
End Sub
Friend Sub FriendMethod () ' Friend
End Sub
Private Sub PrivateMethod () ' Private
```

```
End Sub

End Class
```

على عكس التركيبات، يمكن للفئات أن لا تحتوي على أية أعضاء:

ممكن جدا '

```
Class PersonClass

End Class
```

رسالة خطأ '

```
Structure PersonStructure

End Structure
```

```
Partial Class _Default
    Inherits System.Web.UI.Page

    Class Person
        Public Name As String
        Public Age As Integer
        Public Address As String
        Sub SetValues(ByVal PersonName As String, _
        ByVal PersonAge As Integer, ByVal PersonAddress As St
            Name = PersonName
            Age = PersonAge
            Address = PersonAddress
        End Sub
        Sub print()
            MsgBox("hello " & Name)
        End Sub
    End Class

    Protected Sub Page_Load(ByVal sender As Object, ByVal e A
        Dim obj As New Person()
        obj.SetValues(" ahmad", 29, "jordan")
        obj.print()
    End Sub
End Class
```

تمرين :-

اعمل: class car

تحتوي على :

1. نوع و رقم السيارة وثمنها ولونها, علما بأن الثمن private
2. وتحتوي اقتران يطبع جميع المكونات
3. و اقتران يدخل جميع قيم المتغيرات من خلاله

استخدام الخصائص (Propretes)

```
Class PersonClass

Property BirthDate() As Date '
```

خاصيتي من النوع Date

```
End Property
End Class
```

يمكنك إضافة اجرائين في هذه الخاصية:

– **الإجراء الأول هو Get والذي يتم استدعائه عند قيام بقراءة قيمة الخاصيةز**

– **أما الإجراء Set فسيتم استدعائه عند إسناد قيمة جديدة مستخدم الفئة**

- **كما يفضل استخدام متغير وسيط خاص Private يحمل قيمة الخاصية**

متغير وسيط يحمل قيمة الخاصية '

```
Private m_BirthDate As Date
```

```
Property BirthDate() As Date
Get
Return m_BirthDate
End Get
```

' Value لا بد من ان تطابق نوع قيمة الخاصية مع الوسيطة

' ByVal كما يشترك ارسالها بالقيمة Date وهو

```
Set(ByVal Value As Date)
m_BirthDate = Value
End Set
End Property
```

كيفية استدعاء الخصائص(Proprties):

```
Dim a As New PersonClass()
a.BirthDate = #1/1/1903#
Response.Write(a.BirthDate)
```

مثال :

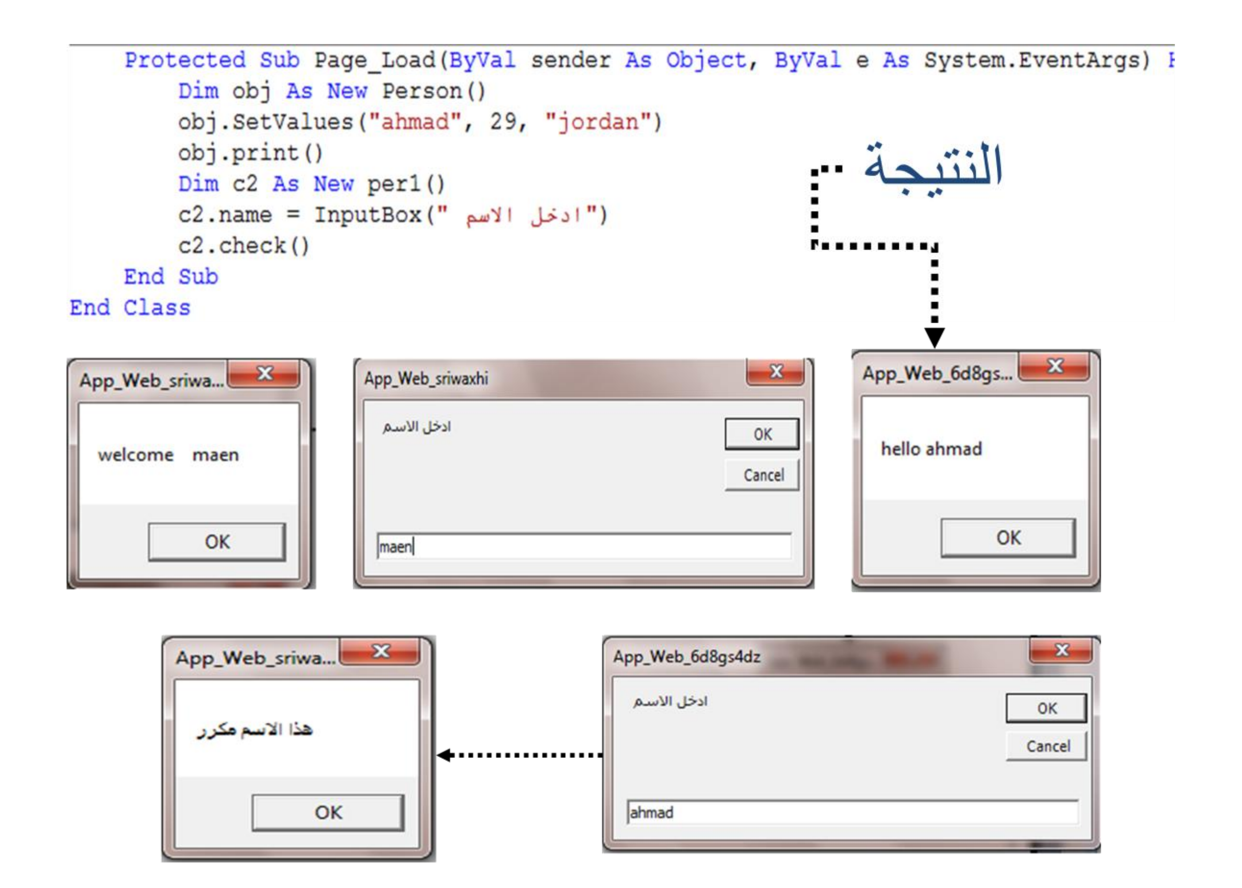
```
Partial Class _Default
    Inherits System.Web.UI.Page
    Class pat
        Private msFirstName As String
        Public Property FirstName() As String
            Get
                Return msFirstName
            End Get
            Set(ByVal Value As String)
                msFirstName = Value
            End Set
        End Property
    End Class

    Protected Sub Page_Load(ByVal sender As O
        Dim Pat1 As New pat()
        Pat1.FirstName = "Bob"
        MsgBox(Pat1.FirstName)
    End Sub
End Class
```

المتغيرات الخاصة وامكانية الوصول اليها :

```
Partial Class _Default
    Inherits System.Web.UI.Page
    Class Person
        Public Name As String
        Private Age As Integer
        Public Address As String
        Sub SetValues(ByVal PersonName As String, _
        ByVal PersonAge As Integer, ByVal PersonAddress As String)
            Name = PersonName
            Age = PersonAge
            Address = PersonAddress
        End Sub
        Sub print()
            MsgBox("hello " & Name)
        End Sub
    End Class
    Class per1
        Public name As String
        Public age As Integer
        Public Sub check()
            Dim o As New Person()
            If name = o.Name Then
                MsgBox("هذا الاسم مكرر")
            Else
                MsgBox("welcome" & name)
            End If
        End Sub
    End Class
```

مثال:



Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Dim obj As New Person()
obj.SetValues("ahmad", 29, "jordan")
obj.print()
Dim c2 As New per1()
c2.name = InputBox(" ادخل الاسم")
c2.check()
End Sub
End Class
النتيجة
App_Web_sriwa...
welcome maen
OK
App_Web_sriwaxhi
ادخل الاسم
OK
Cancel
maen
App_Web_6d8gs...
hello ahmad
OK
App_Web_sriwa...
هذا الاسم مكرر
OK
App_Web_6d8gs4dz
ادخل الاسم
OK
Cancel
ahmad

مثال :-

File Edit View Website Build Debug Tools Window Help Full Screen

Default.aspx.vb | Default.aspx

(Page Events) | Load

```vb
        Public Address As String
        Sub SetValues(ByVal PersonName As String, _
        ByVal PersonAge As Integer, ByVal PersonAddress As String)
            Name = PersonName
            Age = PersonAge
            Address = PersonAddress
        End Sub
        Sub print()
            MsgBox("hello " & Name)
        End Sub
    End Class
    Class per1
        Public name As String
        Public age As Integer
        Public Sub check(ByRef o As Person)

            If name.Equals(o.Name) Then
                MsgBox("هذا الاسم مكرر")
            Else
                MsgBox("welcome     " & name)
            End If
        End Sub
    End Class
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me
        Dim obj As New Person()
        obj.SetValues("ahmad", 29, "jordan")
        obj.print()
        Dim c2 As New per1()
        c2.name = InputBox(" ادخل الاسم")
        c2.check(obj)
    End Sub
End Class
```

Ready   Ln 32   Col 45   Ch 45   INS

مثال :-

```
Partial Class _Default
    Inherits System.Web.UI.Page
    Class Person
        Public Name As String
        Private Age As Integer
        Public Address As String
        Sub SetValues(ByVal PersonName As String, _
        ByVal PersonAge As Integer, ByVal PersonAddress As String)
            Name = PersonName
            Age = PersonAge
            Address = PersonAddress
        End Sub
        Sub print()
            MsgBox("hello " & Name)
        End Sub
    End Class

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As Sys
        Dim obj1 As New Person()
        obj1.SetValues("ahmad", 29, "jordan")
        obj1.print()
        Dim obj2 As New Person()
        obj2.Name = "ali"
        'obj2.age = 34 erorr age is private we cannot access
        obj2.Address = "syria"
        obj2.print()
    End Sub
End Class
```

الكائنات المدرجة داخل بعضها البعض :-

```
    Class school
        Public classes As String
        Public nstd As Integer
        Dim st As New std()
        Sub setvalue()
            classes = InputBox("ادخل اسم الصف")
            nstd = InputBox("ادخل عدد الطلاب")
            Dim n As String = InputBox("ادخل اسم ")
            Dim m As Integer = InputBox("ادخل العلامة")
            st.setvalue(n, m)
        End Sub
        Class std
            Public name As String
            Public mark As Integer
            Public Sub setvalue(ByVal n As String, ByVal m As Integer)
                name = n
                mark = m
            End Sub
        End Class
        Public Sub print()
            MsgBox(st.name & "  your mark is  " & st.mark)
        End Sub
    End Class
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles
        Dim sc As New school()
        sc.setvalue()
        sc.print()
    End Sub
End Class
```

الوحدة السابعة:

مهام الكائنات:-

المصفوفات ARRAY:

يتوقع من الطالب في هذه الوحدة ان يكون قادرا على :

- التعرف على Array List
- مثال على Array List
- المزيد من المعلومات عن Array List
- في مثال فعلي Hash Table
- التعرف على Hash Table
- اضافة العناصر وتغييرها بصورة مباشرة في hash Table

المزيد من المعلومات عن Hash Table

*How to important Array List in VB.NET?

Is one of the most flexible data structure from ***ArrayList*** VB.NET Collections.

- Array List contains a simple list of values.

- Very easily we can add , insert , delete , view etc.. to do with Array List.

- It is very flexible because we can add without any size information , that is it grow dynamically and also shrink.

*** Important functions in Array List**

-Add : Add an Item in an ArrayList

-Insert : Insert an Item in a specified position in an ArrayList

-Remove : Remove an Item from ArrayList

-RemoveAt: remeove an item from a specified position

-Sort : Sort Items in an ArrayList

***How to Add an Item in an ArrayList?**

- *Syntax : ArrayList.add(Item)*

- *Item : The Item to be add the ArrayList*

= Dim ItemList As New ArrayList()

=ItemList.Add("Item4")

***How to Insert an Item in an ArrayList?**

)- Syntax : ArrayList.insert(index,item

index : The position of the item in an ArrayList

- *Item : The Item to be add the ArrayList*

- *ItemList.Insert(3, “item6”)*

*** How to Remove an item from arrayList?**

)Syntax : ArrayList.Remove(item−

Item : The Item to be removed the ArrayList−

- *ItemList.Remove("item2")*

***How to remove an item in a specified position from an Array list**

*– **Syntax : ArrayList.RemoveAt(index)***

- index : the position of an item to remove from an ArrayList

– ItemList.RemoveAt(2)

***How to sort ArrayList**

Syntax : ArrayList.Sort( )

مثال :

```
Partial Class _Default
    Inherits System.Web.UI.Page

    Private Sub Button1_Click(ByVal sender As Sy
        Dim i As Integer
        Dim ItemList As New ArrayList()
        ItemList.Add("Item4")
        ItemList.Add("Item5")
        ItemList.Add("Item2")
        ItemList.Add("Item1")
        ItemList.Add("Item3")
        MsgBox("Shows Added Items")
        For i = 0 To ItemList.Count - 1
            MsgBox(ItemList.Item(i))
        Next
        'insert an item
        ItemList.Insert(3, "Item6")
        'sort itemms in an arraylist
        ItemList.Sort()
        'remove an item
        ItemList.Remove("Item1")
        'remove item from a specified index
        ItemList.RemoveAt(3)
        MsgBox("Shows final Items the ArrayList"
        For i = 0 To ItemList.Count - 1
            MsgBox(ItemList.Item(i))
        Next
    End Sub
End Class
```

